Języki, automaty i obliczenia

Wykład 10: Maszyny Turinga

Sławomir Lasota

Uniwersytet Warszawski

29 kwietnia 2015

Plan

Maszyny Turinga

(Niedeterministyczna) *maszyna Turinga* $\mathcal{M} = (A, Q, q_0, F, T, \mathbb{B}, \delta)$

- A – alfabet wejściowy
- Q – zbiór stanów
- $q_0 \in Q$ – stan początkowy
- $F \subseteq Q$ – stany akceptujące (bez u.o. jeden stan akceptujący)
- T – alfabet taśmowy, $A \subseteq T$
- $\mathbb{B} \in T - A$ – symbol pusty (ang. blank)
- $\delta \subseteq Q \times T \times Q \times T \times \{\leftarrow, \circlearrowleft, \rightarrow\}$ – relacja przejścia

Co oznacza przejście?

$(q, a, q', a', k) \in \delta$: zmień stan z q na q', czytaj a, zapisz a', zmień pozycję wg. k

Konfiguracja początkowa maszyny $\mathcal{M}$ wygląda tak:

$\longleftarrow \quad q \quad \longrightarrow$

∇ (nad komórką z pierwszym symbolem a)

...	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	a	b	a	a	b	a	b	$\mathbb{B}$	$\mathbb{B}$	...

Przykład

- $A = \{a, b\}$
- $Q = \{\mathrm{start}, \mathrm{start}_a, \mathrm{start}_b, \mathrm{ret}, \mathrm{go}, \mathrm{go}_a, \mathrm{go}_b, \mathrm{go}'_a, \mathrm{go}'_b, \mathrm{ret}', \mathrm{check}, \mathrm{ok}\}$
- $q_0 = \mathrm{start}$
- $F = \{\mathrm{ok}\}$
- $T = A \cup \{\mathbb{B}, \#\}$
- relacja przejścia (na następnym slajdzie)

$$\delta : Q \times T \to \mathcal{P}(Q \times T \times \{\leftarrow, \circlearrowleft, \rightarrow\})$$

Pytanie

Jaki język rozpoznaje ta maszyna?

Przykład (relacja przejścia)

	a	b	$\mathbb{B}$	$\#$
start	$(\text{start}_a, \#, \rightarrow)$	$(\text{start}_b, \#, \rightarrow)$	$(\text{ok}, \mathbb{B}, \circlearrowleft)$	
start_a	$(\text{start}_a, a, \rightarrow)$ $(\text{ret}, \#, \leftarrow)$	$(\text{start}_a, b, \rightarrow)$	?	
start_b	$(\text{start}_b, a, \rightarrow)$	$(\text{start}_b, b, \rightarrow)$ $(\text{ret}, \#, \leftarrow)$	?	
ret	$(\text{ret}, a, \leftarrow)$	$(\text{ret}, b, \leftarrow)$		$(\text{go}, \#, \rightarrow)$
go	$(\text{go}_a, \#, \rightarrow)$	$(\text{go}_b, \#, \rightarrow)$		$(\text{check}, \#, \circlearrowleft)$
go_a	$(\text{go}_a, a, \rightarrow)$	$(\text{go}_a, b, \rightarrow)$		$(\text{go}'_a, \#, \rightarrow)$
go_b	$(\text{go}_b, a, \rightarrow)$	$(\text{go}_b, b, \rightarrow)$		$(\text{go}'_b, \#, \rightarrow)$
go'_a	$(\text{ret}', \#, \leftarrow)$	?	?	$(\text{go}'_a, \#, \rightarrow)$
go'_b	?	$(\text{ret}', \#, \leftarrow)$	?	$(\text{go}'_b, \#, \rightarrow)$
ret'	$(\text{ret}, a, \leftarrow)$	$(\text{ret}, b, \leftarrow)$	$(\text{check}, \mathbb{B}, \rightarrow)$	$(\text{ret}', \#, \leftarrow)$
check	?	?	$(\text{ok}, \mathbb{B}, \circlearrowleft)$	$(\text{check}, \#, \rightarrow)$

Przykład (język)

$$L(\mathcal{M}) \;=\; \{w\,w \;:\; w \in A^*\}$$

Pytanie

Jak zmodyfikować maszynę $\mathcal{M}$, aby obliczała funkcję:

$$w \;\mapsto\; w\,w?$$

Taśma

...	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	a	b	a	a	b	a	b	$\mathbb{B}$	$\mathbb{B}$	...

- Taśma jest nieskończona w obydwie strony.
- Taśma reprezentuje pamięć maszyny: nie nieskończoną, ale dowolnie dużą skończoną.
- Bez u.o. możemy założyć, że maszyna nigdy nie zapisuje symbolu $\mathbb{B}$; wtedy $\mathbb{B}$ oznacza „nieużywane” pozycje taśmy.
- Poza obszarem używanym przez maszynę, na wszystkich pozycjach taśmy jest symbol $\mathbb{B}$.
- Nieistotne, na której pozycji taśmy zaczyna się obszar używany przez maszynę (przesunięcie tego obszaru w prawo lub w lewo nic istotnego nie zmienia).

Zapisując konfiguracje, pomijamy nieskończenie wiele symboli $\mathbb{B}$ poza obszarem odwiedzonym przez maszynę, i poza słowem wejściowym:

$\longleftarrow$ q $\longrightarrow$

∇

...	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	a	b	a	a	b	a	b	$\mathbb{B}$	$\mathbb{B}$	...

$$(q, a)\, b\, a\, a\, b\, a\, b \qquad \text{albo} \qquad q\, a\, b\, a\, a\, b\, a\, b$$

Formalnie, *konfiguracja* maszyny $\mathcal{M}$ to

$$c \;=\; w\, q\, w' \in T^* Q\, T^*$$

Umowa

$q\, a\, b\, a\, a\, b\, a\, b \;=\; \mathbb{B}\, q\, a\, b\, a\, a\, b\, a\, b \;=\; q\, a\, b\, a\, a\, b\, a\, b\mathbb{B} \;=\; \mathbb{B}\mathbb{B}\, q\, a\, b\, a\, a\, b\, a\, b\mathbb{B} \;=\; \ldots$

Konfiguracje początkowa: $c_0 \;=\; q_0\, w$ ($w \in A^*$ to słowo wejściowe)

Konfiguracje akceptujące: $T^* F\, T^* \;=\; \{w\, q\, w' \;:\; q \in F;\; w, w' \in T^*\}$

Język rozpoznawany przez maszynę

Przejścia pomiędzy konfiguracjami $c \longrightarrow_{\mathcal{M}} c'$: $(w, v \in T^*)$

- jeśli $(q, a, q', a', \circlearrowleft) \in \delta$ to

$$w\, q\, a\, v \longrightarrow_{\mathcal{M}} w\, q'\, a'\, v$$

- jeśli $(q, a, q', a', \rightarrow) \in \delta$ to

$$w\, q\, a\, v \longrightarrow_{\mathcal{M}} w\, a'\, q'\, v$$

- jeśli $(q, a, q', a', \leftarrow) \in \delta$ to

$$w\, b\, q\, a\, v \longrightarrow_{\mathcal{M}} w\, q'\, b\, a'\, v$$

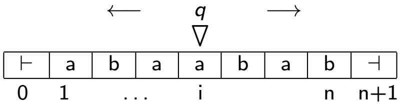
Umowa

$q\,a\,b\,a\,a\,b\,a\,b \;=\; \mathbb{B}\,q\,a\,b\,a\,a\,b\,a\,b \;=\; q\,a\,b\,a\,a\,b\,a\,b\,\mathbb{B} \;=\; \mathbb{B}\mathbb{B}\,q\,a\,b\,a\,a\,b\,a\,b\,\mathbb{B} \;=\; \ldots$

Język rozpoznawany przez maszynę $\mathcal{M}$:

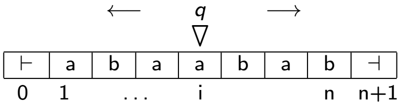
$$L(\mathcal{M}) \;=\; \{w \in A^* \;:\; q_0\, w \longrightarrow^*_{\mathcal{M}} c \text{ dla jakiejś konfiguracji akceptującej } c\}$$

Dwukierunkowe automaty skończone



Automaty dwukierunkowe to...

Dwukierunkowe automaty skończone



Automaty dwukierunkowe to. . . maszyny Turinga, które nie mogą pisać na taśmie.

Języki rekurencyjnie przeliczalne

Klasa języków rozpoznawanych przez maszyny Turinga:

- języki *częściowo* *rozstrzygalne*
- języki *rekurencyjnie przeliczalne*

Hierarchia Chomsky'ego:

(typ 0) języki rekurencyjnie przeliczalne

(typ 1) języki kontekstowe

(typ 2) języki bezkontekstowe

(typ 3) języki regularne

Plan

Przykład (maszyna deterministyczna)

$$L(\mathcal{M}) \;=\; \{a^n b\, a^n \;:\; n \in \mathbb{N}\}$$

	a	b	$\mathbb{B}$	$\#$
$\text{start}_{\rightarrow}$	$(\text{cont}_{\rightarrow}, \#, \rightarrow)$	$(\text{check}, b, \rightarrow)$	$(\text{nok}, \mathbb{B}, \circlearrowleft)$	$(\text{nok}, \#, \circlearrowleft)$
$\text{cont}_{\rightarrow}$	$(\text{cont}_{\rightarrow}, a, \rightarrow)$	$(\text{cont}_{\rightarrow}, b, \rightarrow)$	$(\text{start}_{\leftarrow}, \mathbb{B}, \leftarrow)$	$(\text{start}_{\leftarrow}, \#, \leftarrow)$
$\text{start}_{\leftarrow}$	$(\text{cont}_{\leftarrow}, \#, \leftarrow)$	$(\text{nok}, b, \circlearrowleft)$	$(\text{nok}, \mathbb{B}, \circlearrowleft)$	$(\text{nok}, \#, \circlearrowleft)$
$\text{cont}_{\leftarrow}$	$(\text{cont}_{\leftarrow}, a, \leftarrow)$	$(\text{cont}_{\leftarrow}, b, \leftarrow)$	$(\text{nok}, \mathbb{B}, \circlearrowleft)$	$(\text{start}_{\rightarrow}, \#, \rightarrow)$
check	$(\text{nok}, a, \circlearrowleft)$	$(\text{nok}, b, \circlearrowleft)$	$(\text{nok}, \mathbb{B}, \circlearrowleft)$	$(\text{ok}, \#, \circlearrowleft)$

Maszyna deterministyczna:

$$\delta : (Q - \{\text{ok}, \text{nok}\}) \times T \rightarrow Q \times T \times \{\leftarrow, \circlearrowleft, \rightarrow\}$$

Przykład (maszyna deterministyczna)

$$L(\mathcal{M}) = \{a^n b\, a^n \ :\ n \in \mathbb{N}\}$$

	a	b	$\mathbb{B}$	$\#$
$\text{start}_\rightarrow$	$(\text{cont}_\rightarrow, \#, \rightarrow)$	$(\text{check}, b, \rightarrow)$	$(\text{nok}, \mathbb{B}, \circlearrowleft)$	$(\text{nok}, \#, \circlearrowleft)$
$\text{cont}_\rightarrow$	$(\text{cont}_\rightarrow, a, \rightarrow)$	$(\text{cont}_\rightarrow, b, \rightarrow)$	$(\text{start}_\leftarrow, \mathbb{B}, \leftarrow)$	$(\text{start}_\leftarrow, \#, \leftarrow)$
$\text{start}_\leftarrow$	$(\text{cont}_\leftarrow, \#, \leftarrow)$	$(\text{nok}, b, \circlearrowleft)$	$(\text{nok}, \mathbb{B}, \circlearrowleft)$	$(\text{nok}, \#, \circlearrowleft)$
$\text{cont}_\leftarrow$	$(\text{cont}_\leftarrow, a, \leftarrow)$	$(\text{cont}_\leftarrow, b, \leftarrow)$	$(\text{nok}, \mathbb{B}, \circlearrowleft)$	$(\text{start}_\rightarrow, \#, \rightarrow)$
check	$(\text{nok}, a, \circlearrowleft)$	$(\text{nok}, b, \circlearrowleft)$	$(\text{nok}, \mathbb{B}, \circlearrowleft)$	$(\text{ok}, \#, \circlearrowleft)$

Maszyna deterministyczna:

$$\delta : (Q - \{\text{ok}, \text{nok}\}) \times T \rightarrow Q \times T \times \{\leftarrow, \circlearrowleft, \rightarrow\}$$

Pytanie

Czy maszynę z poprzeniego przykładu można zdeterminizować?

Determinizacja?

Deterministyczna *maszyna Turinga* $\mathcal{M} = (A, Q, q_0, q_{\text{tak}}, q_{\text{nie}}, T, \mathbb{B}, \delta)$

- $q_{\text{tak}} \in Q$ – stan akceptujący
- $q_{\text{nie}} \in Q$ – stan odrzucający
- $\delta : (Q - \{q_{\text{tak}}, q_{\text{nie}}\}) \times T \to Q \times T \times \{\leftarrow, \circlearrowleft, \rightarrow\}$

Pytanie

Czy dla każdej maszyny Turinga istnieje równoważna maszyna deterministyczna?

Plan

Warianty maszyn Turinga

Maszyny z:

- taśmą jednostronnie nieskończoną
- wieloma taśmami
- „taśmą" wielowymiarową
- ...

Maszyny z taśmą jednostronnie nieskończoną

$\vdash$	a	b	a	a	b	a	b	$\mathbb{B}$	$\mathbb{B}$	...

Pytanie

Czy maszyna z taśmą jednostronnie nieskończoną potrafi symulować maszynę z taśmą dwustronnie nieskończoną?

Maszyny z taśmą jednostronnie nieskończoną

$\vdash$	a	b	a	a	b	a	b	$\mathbb{B}$	$\mathbb{B}$	...

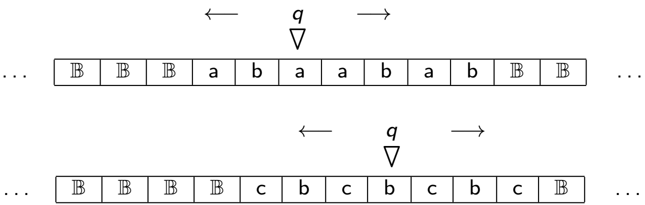
Pytanie

Czy maszyna z taśmą jednostronnie nieskończoną potrafi symulować maszynę z taśmą dwustronnie nieskończoną?

...	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	a	b	a	a	b	a	b	$\mathbb{B}$	$\mathbb{B}$	...

	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$			
$\vdash$	a	b	a	a	b	a	b	$\mathbb{B}$	$\mathbb{B}$	...

Maszyny wielotaśmowe



$$\delta \subseteq Q \times T^2 \times Q \times T^2 \times \{\leftarrow, \circlearrowleft, \rightarrow\}^2$$

Pytanie

Czy maszyna jednotaśmowa potrafi symulować maszynę wielotaśmową?

Maszyny wielotaśmowe

$\longleftarrow \quad q \quad \longrightarrow$

∇

...	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	a	b	a	a	b	a	b	$\mathbb{B}$	$\mathbb{B}$	...

$\longleftarrow \quad q \quad \longrightarrow$

∇

...	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	c	b	c	b	c	b	c	$\mathbb{B}$	...

$$\delta \subseteq Q \times T^2 \times Q \times T^2 \times \{\leftarrow, \circlearrowleft, \rightarrow\}^2$$

Pytanie

Czy maszyna jednotaśmowa potrafi symulować maszynę wielotaśmową?

$\longleftarrow \quad q \quad \longrightarrow$

∇

...			c	a	b	$\bar{a}$	a	b	a	b	$\mathbb{B}$		...
	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	c	b	c	$\underline{b}$	c	b	c	$\mathbb{B}$	

Maszyny z taśmą dwuwymiarową

...

$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$
$\mathbb{B}$	$\mathbb{B}$	a	b	b	c	q	a	a	a	$\mathbb{B}$	$\mathbb{B}$
$\mathbb{B}$	a	b	a	a	b	∇	a	a	b	$\mathbb{B}$	$\mathbb{B}$
$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	a	b	a	a	b	a	b	$\mathbb{B}$	$\mathbb{B}$
$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	c	b	c	b	c	b	c	$\mathbb{B}$
$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$

... (left) ... (right)

...

$$\delta \subseteq Q \times T \times Q \times T \times \{\leftarrow, \circlearrowleft, \rightarrow, \uparrow, \downarrow\}$$

Pytanie

Czy maszyna z taśmą jednowymiarową potrafi symulować maszynę z taśmą dwuwymiarową?

Maszyny z taśmą dwuwymiarową

...

$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$
$\mathbb{B}$	$\mathbb{B}$	a	b	b	c	q	a	a	a	$\mathbb{B}$	$\mathbb{B}$
$\mathbb{B}$	a	b	a	a	b	∇	a	a	b	$\mathbb{B}$	$\mathbb{B}$
$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	a	b	a	a	b	a	b	$\mathbb{B}$	$\mathbb{B}$
$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	c	b	c	b	c	b	c	$\mathbb{B}$
$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$	$\mathbb{B}$

... (po lewej i prawej stronie) ...

...

$$\delta \subseteq Q \times T \times Q \times T \times \{\leftarrow, \circlearrowleft, \rightarrow, \uparrow, \downarrow\}$$

Pytanie

Czy maszyna z taśmą jednowymiarową potrafi symulować maszynę z taśmą dwuwymiarową?

$\vdash$	a	#	a	b	b	c	b	a	b	b	#	a	a	a	c	b	...

Plan

Języki jako problemy decyzyjne

Języki będziemy utożsamiać z *problemami decyzyjnymi* (*zadaniami obliczeniowymi*).

Przykład

Dane wejściowe:	graf skierowany G
Wynik:	rozstrzygnąć, czy G ma cykl Hamiltona?

Graf można opisać jako słowo nad $A = \{0, 1, \#\}$, np. 110#101#001

Ogólnie, język $L \subseteq A^*$ można utożsamić z następującym zadaniem obliczeniowym:

Dane wejściowe:	$w \in A^*$
Wynik:	rozstrzygnąć, czy $w \in L$?

Teza Churcha-Turinga:

języki rozpoznawane (problemy obliczane) przez maszyny Turinga	=	problemy, dla których istnieje efektywny algorytm, przy założeniu nieograniczonych zasobów

albo:

maszyny Turinga	=	komputery

Wątpliwości:

- nieskończoność taśmy?
- alfabet stosowy większy niż rozmiar dysku?
- liczba stanów większa niż rozmiar dysku?
- niedeterminizm?

Język uniwersalny (problem stopu)

Problem stopu

Dane: maszyna Turinga $\mathcal{M}$ nad alfabetem A i słowo $w \in A^*$

Wynik: czy $w \in L(\mathcal{M})$?

Maszynę $\mathcal{M}$ można opisać jako słowo

$$\text{kod}_{\mathcal{M}} \in \{0, 1, \#\}^*,$$

na przykład

$$00\#000000\#001000\#000011\#000\#\#000100\#001\#010000\#100\#001\#\#\ldots$$

Nie ma istotnej różnicy między programem a daną...

Pytanie

Czy alfabet $\{0, 1, \#\}$ można zmiejszyć?

Język uniwersalny i maszyna uniwersalna

Bez u.o. możemy ograniczyć się do maszyn, których alfabet taśmowy $T = \{0, 1, \mathbb{B}\}$. Faktycznie, słowo nad alfabetem $T - \{\mathbb{B}\} = \{a, b, c, d\}$ można zapisać jako słowo nad $\{0, 1\}$, na przykład:

$$a\,b\,c\,a\,d \;=\; 1000 \quad 0100 \quad 0010 \quad 1000 \quad 0001$$

Język uniwersalny (problem stopu):

$$\begin{array}{rcl} \{\ \ (\mathcal{M}, w) & : & w \in L(\mathcal{M})\} \\ \{\text{kod}_{\mathcal{M}}\,\$\,w & : & w \in L(\mathcal{M})\} \ \subseteq\ \{0, 1, \#, \$\}^* \end{array}$$

Twierdzenie

Język uniwersalny jest rekurencyjnie przeliczalny.

Dowód:

Uniwersalna maszyna Turinga:

$000100\,\$$

$00\#000000\#001000\#000011\#000\#\#000100\#001\#010000\#100\#001\#\#\dots$

$\dots\$\,00101010101\overline{1}111101011$

Modele obliczeń równoważne maszynom Turinga

- gramatyki typu 0
- automaty wielostosowe
- maszyny (automaty) licznikowe
- automaty z kolejką
- ...
- maszyny RAM (ang. random access machines) albo maszyny rejestrowe
- rachunek λ
- ...

W następnym odcinku:

Obliczalność i nieobliczalność